

COL362 Assignment 1

Does 3 level architecture solve all the problems?

It does manage to include separation of concerns, providing an extra layer of security and abstraction for the users. However, it may increase the size of the codebase and lead to issues if there are problems with the communication between the 2 tiers (latency issues etc due to physical separation).

How can 3 level architecture be implemented? What transformations and data structures are needed?

The most common implementation of the 3-tier architecture can be seen in modern web applications, for example LAMP(Linux, Apache, MySQL, PHP) and MEAN(Mongo, Express, Angular, NodeJS) stacks. Here, the business logic is contained within a PHP/ NodeJS Application layer that communicates with the database layer (MySQL/MongoDB) to provide an API service. The “frontend” (a web page in angularJS or maybe a native GUI application, for example), consumes data from this API and displays it to the user.

Reference : [https://msdn.microsoft.com/en-us/library/windows/desktop/ms685068\(v=vs.85\).aspx](https://msdn.microsoft.com/en-us/library/windows/desktop/ms685068(v=vs.85).aspx) and https://en.wikipedia.org/wiki/Multitier_architecture (refer to section on web development usage)

MDA(Model Driven Architecture) based Transformations used to generate the lower level representations. Most of the data is stored in the form of B-trees

Does 3 level architecture support more than 1 programming language?

Yes. The different layers can be built on top of any programming language platform (however, the application layer should have an ODBC so it can do database operations). They can communicate with each other typically using a language independent format like JSON or XML.

How is data independence achieved?

When the various tiers in the architecture can be mixed and matched (as long as compatibility with the communication layer can be maintained), the tiers can be

independent of each other. This is called physical data independence. For example, NodeJS middleware can communicate with POSTGRESQL or MongoDB and it won't matter to the frontend application, as long as the API returns JSON that can be consumed.